



HAL
open science

Novel Slice Admission Control Scheme with Overbooking and Dynamic Buyback Process

Solomon Yese, Sara Berri, Arsenia Chorti

► To cite this version:

Solomon Yese, Sara Berri, Arsenia Chorti. Novel Slice Admission Control Scheme with Overbooking and Dynamic Buyback Process. IEEE Conference on Network Function Virtualization and Software Defined Networks, Nov 2023, Dresden, Germany. ⟨hal-04246439⟩

HAL Id: hal-04246439

<https://hal.science/hal-04246439v1>

Submitted on 17 Oct 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



HAL Authorization

Novel Slice Admission Control Scheme with Overbooking and Dynamic Buyback Process

Solomon Yese, Sara Berri, Arsenia Chorti

ETIS, UMR 8051 CY Cergy Paris University, ENSEA, CNRS F-95000
solomon.yese@ensea.fr, sara.berri@ensea.fr, arsenia.chorti@ensea.fr

Abstract—The emergence of 5G systems brought to fore the importance of network slicing (NS) as it allows infrastructure providers (InPs) to create logical networks (slices) and virtually share network resources to their tenants. However, due to the limited resources of the InP, the resource management algorithms like resource allocation and admission control are required to ensure efficient management of these scarce resources. Indeed, admission control algorithms play a critical role of regulating access to the network, by determining whether a slice request should be accepted or not with respect to some standards such as maximizing the InP’s revenue and maintaining service level agreements (SLAs). In this paper, we propose an admission control algorithm that employs the concept of overbooking to admit slice requests beyond the InP’s nominal available resources. Moreover, we employ a dynamic queue adaption priority, step-wise pooling and dynamic buyback price mechanism to ensure efficient and profitable admission decision for the InP. We assess the performance of the proposed algorithm against state of the art (SOTA) solution considering different priority schemes. The results show that the proposed solution outperforms the SOTA solution as it yields *i)* higher revenue, *ii)* lower buyback cost and *iii)* higher net revenue for the InP while still maintaining a marginally higher slice acceptance rate.

Index Terms—Admission Control, Network Slicing, 5G/B5G, Dynamic Priority, Dynamic Pricing.

I. INTRODUCTION

The fifth generation (5G) and beyond of mobile networks promise to provide a hundred times (10 gigabit/seconds) faster data delivery than the current networks [1]. Therefore, it is expected to simultaneously serve large number of devices with multiple heterogeneous services having diverse quality of service (QoS) requirements such as throughput, delay, reliability, etc. [2]. To provide efficient service delivery for these diverse requirements, 5G architecture provides support through infrastructural sharing or network slicing which allows several logical networks called slices to share the resources of a single physical network [3], [4].

Due to the limited nature of network infrastructure resources and the quality guarantees required by network slices, mechanisms such as admission control are needed before new slice requests are accepted. This is to ensure that the new slices are only accepted if their service guarantees can be met by the network without adversely affecting the quality guarantees for existing slices. For this purpose, a slice admission controller must make decisions that maximize its utility, such as the rev-

enue, based on the current system occupation and its expected long-term evolution [5].

Over the years, several admission control algorithms have been proposed for either revenue maximization, QoS control assurance (service level agreements (SLA)), or improve resource utilization. More recently, some works have attempted to simultaneously achieve two or more of these objectives [6]-[8] by employing the concept of overbooking and resource sharing. This allows the InP to accept requests beyond its nominal available resources by buying back resources from existing tenants when there are insufficient resources to accept new requests. However, they buyback all idle resources without recourse to the resource requirements of the waiting requests thus, increasing the provider’s buyback cost and decreasing the net revenue. Similarly, they define priorities that mostly depend on slice type that provide short term revenue gains but lower long term revenue. Therefore, it is necessary to design novel algorithms for network slicing [9] with an adaptive buyback process. In this paper, we study the admission control problem and aim to maximize the revenue of infrastructure providers and minimize their buyback cost. The key contributions of this paper are as follows:

- We formulate the problem as a multi-objective problem to simultaneously maximize the revenue and minimize the buyback cost of the InP.
- We propose a heuristic to yield a near-optimal solution, different slice priority schemes to guide the decision process, and a dynamic and realistic buyback price model. Moreover, the algorithm employs a dynamic step-wise pooling mechanism to buyback only the needed resources and thus reduce the cost.
- We show through extensive simulations the impact of the proposed solution on the InP’s revenue, buyback cost, net revenue and slice acceptance rate in comparison with the state of the art solution.

The organization of the rest of the paper is as follows. Section II presents the related works. In Section III, we introduce the network model and the problem formulation. Section IV introduces the proposed algorithm while Section V captures the performance evaluation of the proposed algorithm against the SOTA. Finally, Section VI concludes the paper.

II. RELATED WORKS

In this section, we present some key related literature on admission control in 5G and B5G network slicing.

A squatting and kicking (SKM) scheme is proposed in [6] to improve resource utilization. It partitions and reserves resources according to the various priority classes while borrowing from existing slices when there are no resources to admit new requests. The scheme improves resource utilization and increases acceptance rate of high priority slices. However, the scheme's squatting and kicking of existing slices without recourse to the needed resources will reduce resource utilization of the network. It will also lead to the violation of SLAs when there is total kicking.

In [7], a reinforcement-learning (RL) based admission control is proposed to maximize profit. The RL algorithm improves revenue but will not perform optimally when the state space grows. More so, the computational resources required will lead to increased cost for operators and consequently tenants and users.

A RL with Deep Q networks based algorithm is proposed in [8] to maximize long term revenue. It processes requests in the waiting queue on a potential revenue basis, and employs a dynamic pricing model that defines the revenue accrued from admitting a request as a product of the request's urgency, duration and sum of the product of the supply/demand function and base price fair. The proposed scheme improves the InP's revenue but reduces acceptance rate. More so, the pricing model employed is not realistic in real network scenario where prices changes do not happen on the fly.

In [9], a deep reinforcement learning based admission control algorithm is proposed to increase revenue and minimize cost of deployment. To maximize revenue, it accepts slices with more profitability in the long run and minimize the cost of deployment in open radio access network (O-RAN) enabled network by efficient placement of the slices. The scheme improves operator revenue and reduces cost. However, it is not suitable for deployment in heterogeneous 5G/B5G use cases scenario as some may never get accepted due to the profit-centric nature of the scheme. That is, in heterogeneous slice scenarios, mMTC slices for example with low revenue generating tendencies may greatly suffer rejection.

In [10], a prediction and RL based overbooking (PRLOV) to maximize revenue and improve resource utilization is proposed. The PRLOV considers low priority-low revenue (Elastic) and high priority- high revenue (Inelastic) slices. It employs the concept of overbooking and predicts the future resource needs of ongoing elastic slices by employing a slice forecasting block (SRF) using the long short-term memory (LSTM) approach to dynamically adjust the overbooking amount. The scheme increases revenue for InPs when the arrival rate of inelastic slices is high and improves resource utilization. However, the scheme's overbooking without recourse to the needed resources may lead to reduced revenue for the InP in the event that the required resources are less than the overbooked resources. It also lowers acceptance rate of elastic slice requests even at low arrival rate of requests due to the reservation used.

In [11], a prioritized slice admission control considering fairness (PSACCF) is proposed to improve fairness. The PSACCF

is a multi-queue based online algorithm based on two parameters: priority level and the threshold of acceptable fairness. It employs the concept of cumulative service acceptance ratio (CSAR) which enforces priority as a preference instead of the absolute privilege that usually characterizes priority. The scheme improves admission fairness but reduces revenue.

While the existing solutions in [6] - [11] increase the InP's revenue, they incur high buyback cost when there is high traffic of slice requests thus, reducing the InP's net revenue. Moreover, these schemes employ priorities that lead to high short term revenue but low long term revenue and high rejection rate for low priority requests. In contrast to these works, we propose a solution that seeks to increase both long and short term revenue of the InP while reducing the buyback cost.

III. NETWORK MODEL AND PROBLEM FORMULATION

In this section, we will describe the network model and formulate the problem of admission control for sliced networks as a multi-objective problem.

A. System Model

We consider a scenario with a single infrastructure provider (InP) having a set of resources, namely computing, storage and network resources, to serve slice requests of different types and diverse resource requirements. Indeed, we consider a set of predefined classes of slices with diverse priorities, denoted in the sequel as $\{C_1, \dots, C_6\}$. The different classes $C_1, C_2, C_3, C_4, C_5, C_6$ correspond to homogeneous - URLLC, heterogeneous - URLLC, heterogeneous - eMBB, homogeneous - eMBB, heterogeneous - mMTC and homogeneous - mMTC slice classes, respectively. Homogeneous - URLLC slices, homogeneous - eMBB slices and homogeneous - mMTC slices are the slices that support only URLLC, eMBB and mMTC use cases, respectively. On the other hand, heterogeneous - URLLC slices, heterogeneous - eMBB slices and heterogeneous - mMTC correspond to the slices that support multiple use cases but more of URLLC, eMBB and mMTC, respectively. Moreover, we denote the computing, storage and network capacity of the InP as R_1, R_2 , and R_3 , respectively, and the set of all the slice requests in queue, awaiting admission decision, as $\mathcal{N} = \{1, 2, \dots, N\}$. Each slice request $i \in \mathcal{N}$ has a duration d_i and a set of requirements to be fulfilled defined as follows: computing $r_{i,1}$, storage $r_{i,2}$ and network resources $r_{i,3}$. The sketch of the system model is represented by Fig. 1.

Furthermore, in the proposed model, we consider that the InP might buyback some resources from already accepted slices when there are requests but the resources in the InP's resource bank are not sufficient to admit any of them. We denote by $r'_{i,j}, \forall j \in \{1, 2, 3\}$ the amount of resources that slice i makes available, where $r'_{i,j} \leq r_{i,j}$. Indeed, this creates a win-win situation for the tenant and InP as the tenant will recoup some of its investment by selling unused resources, that it would have lost at the end, and the InP makes some marginal profit while creating greater satisfaction for its clients. In addition, in contrast to existing solutions that unilaterally assume that all accepted slices are willing to sell the idle resources, we

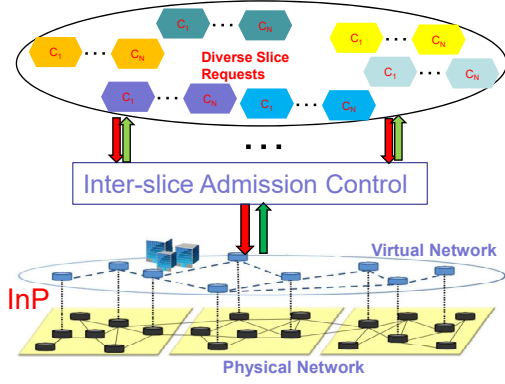


Fig. 1: System Model

introduce the concept of willingness to sell probability to determine the likelihood of a slice being willing to sell its unused resources to the InP. Formally, we define a binary parameter δ_i as the indicator that an admitted request i is a candidate request for buyback ($\delta_i = 1$) or not ($\delta_i = 0$); δ_i can be defined according to relation (1).

$$\delta_i = \begin{cases} 1, & \text{if } P_i \leq \Gamma_i \\ 0, & \text{otherwise.} \end{cases} \quad (1)$$

Where: P_i and Γ_i are the willingness to sell probability and the minimum threshold probability of the admitted slice request i , respectively.

Therefore, the total resources of $R_j, \forall j \in \{1, 2, 3\}$, that can be bought back is defined as :

$$R_j^s = \sum_{i=1}^N \delta_i r'_{i,j}. \quad (2)$$

B. Problem Formulation

In this subsection, we formulate the problem of admission control as a multi-objective problem. Indeed, the main objective is to simultaneously maximize the InP's revenue and minimize the buyback cost, incurred for the recovered resources from the admitted slices, given the request's requirements and the total capacities of the InP in terms of computing, storage and network resources. For this purpose, we formulate the problem as a multi-objective integer linear program. We introduce a binary variable x_i that indicates the successful admission of request i ($x_i = 1$) or not ($x_k = 0$). On the other hand, we define by $p_{i,j}$ and $p'_{i,j}$ the selling unit price and the buyback unit price per unit of the resource $R_j, \forall j \in \{1, 2, 3\}$, and request $i \in \mathcal{N}$, respectively. Note that, in this work, we do not consider additional costs such as the service fees. Indeed, we consider the gross earnings from admitting tenants, because we assume that an InP will not render services at a loss and as such all unit prices set are those that will be profitable to the InP.

Formally, the admission control problem including a buyback scheme is formulated as follows:

$$\begin{cases} U = \max \sum_{i=1}^N \sum_{j=1}^3 x_i r_{i,j} p_{i,j} \\ C = \min \sum_{i=1}^N \sum_{j=1}^3 x_i \delta_i r'_{i,j} p'_{i,j} \end{cases} \quad (3)$$

Subject to:

$$\sum_{i=1}^N x_i r_{i,j} \leq R_j, \quad \forall j \in \{1, 2, 3\}, \quad (4)$$

$$x_i \in \{0, 1\}, \quad \forall i \in \mathcal{N}. \quad (5)$$

The objective in (3) is to maximize the total revenue and minimize the buyback cost that depend on the total admitted requests and the corresponding required resources and prices. The constraints (4) are introduced to ensure that the total amount of distributed resources over the admitted slices cannot exceed the total capacity of the InP. In addition, the constraints (5) indicate the binary decision variables of the problem.

IV. THE PROPOSED ALGORITHM

The objective is to obtain the number of admitted slices which maximizes the total revenue and minimize the buyback price defined by (3) subject to the system capacity and the slices' requirements (4). To solve this problem, in this section, we propose a heuristic algorithm to seek a near-optimal solution. The pseudo-code of the proposed algorithm is presented in Algorithm 1.

Initially, we fix all the parameters including the number of slice requests and their requirements, the InP capacity in terms of computing, storage and network resources, the willingness to sell probability and the threshold, and the different prices. The proposed algorithm sorts the slice requests according to a predefined priority, $\alpha_i \forall i \in \mathcal{N}$, and creates the new ordered set \mathcal{L}^s . This metric, namely $\alpha_i, \forall i \in \mathcal{N}$, will be defined so that the InP's utility (3) is improved as much as possible. We search the requests from top to down according to α_i , and admit the slice if this does not violate the capacity constraint of the InP (4). Once the slice is admitted the total number of available resources at the InP, $R_j, \forall j \in \{1, 2, 3\}$, and the accepted requests list, denoted by \mathcal{L}^a , are updated accordingly. However, if the units of resources in the InP's resource pool is not sufficient to meet the request's requirement, the request is added to the waiting list, denoted by \mathcal{L}^w , and the amount of needed resources to satisfy a maximum number of slice requests is updated $R_j^N, \forall j \in \{1, 2, 3\}$.

After the algorithm finishes the allocation of the InP's resources, it proceeds to determine the candidate requests from the list of accepted requests, that accept to sell some units of their idle resources. We denote the set of candidate requests by \mathcal{L}^c . Indeed, this allows the InP to satisfy waiting requests in \mathcal{L}^w that have not been attended to due to insufficient resources. To determine which requests to buyback from, namely the set \mathcal{L}^c , and the corresponding resources $R_j^s, \forall j \in \{1, 2, 3\}$, we exploit the willingness to sell probability P_i (line 18-line 20). Afterwards, we introduce the incremental/step-wise pooling mechanism based on a sufficiency pre-check mechanism (SPM) as in line 22. Indeed, this is considered to ensure that even when there are much shareable/idle resources, the InPs do not buyback more than is necessary at a particular point in time as this can adversely affect their revenue and willingness to

buyback. Based on this mechanism, if the amount of resources required by all the requests of the waiting list, R_j^N , is less than the sum of the amount of resources remaining in the InP's resource bank and the resources that can be bought back from candidate requests, R_j^s , we employ partial pooling to buyback only the amount of resources needed to meet the demand of requests in the waiting list and accept them (line 23- line 28). Otherwise, we employ full pooling procedure where the InP buys back all the idle resources R_j^s from candidate requests given by (2). Thereafter, the algorithm accepts the slices in the waiting list from top to down according to α_i and update the resources accordingly (line 32- line 35). At last, if a request is still not satisfied, it is considered rejected (line 37).

Algorithm 1: The proposed admission control algorithm.

```

2 Setting:  $R_j, r_{i,j}, P_i, p_{i,j}, \Gamma_i, R_j^s = 0, R_j^N = 0, \mathcal{L}^a = \emptyset,$ 
    $\mathcal{L}^w = \emptyset, \mathcal{L}^c = \emptyset, \mathcal{L}^s = \emptyset.$ 
4 Output:  $x_i, \forall i \in \mathcal{N}$ 
5 begin
7   sort the slice requests in decreasing order of priority
   defined by  $\alpha_i$  and add them to  $\mathcal{L}^s$ .
9   for  $i \in \mathcal{L}^s$  do
10     if  $r_{i,j} \leq R_j, \forall j \in \{1, 2, 3\}$  then
11        $\mathcal{L}^a = \mathcal{L}^a \cup i$  and  $x_i = 1$ 
12        $R_j = R_j - r_{i,j}, \forall j \in \{1, 2, 3\}$ 
13     else
14        $\mathcal{L}^w = \mathcal{L}^w \cup i$ 
15        $R_j^N = R_j^N + r_{i,j}, \forall j \in \{1, 2, 3\}$  /* update
   needed resource */
17   for  $i \in \mathcal{L}^a$  do
18     if  $P_i \leq \Gamma_i$  then
19        $\mathcal{L}^c \cup i$ 
20        $R_j^s = R_j^s + r'_{i,j}, \forall j \in \{1, 2, 3\}$  /* add idle
   resources to shareable
   resources */
22   if  $R_j^N < (R_j + R_j^s), \forall j \in \{1, 2, 3\}$  then
   /* Activate partial pooling */
23     for  $i \in \mathcal{L}^c$  do
24       if  $(R_j + r'_{i,j}) \leq R_j^N, \forall j \in \{1, 2, 3\}$  then
25          $R_j = R_j + r'_{i,j}, \forall j \in \{1, 2, 3\}$ 
26     for  $i \in \mathcal{L}^w$  do
27        $\mathcal{L}^a \cup i$  and  $x_i = 1$ 
28        $R_j = R_j - r_{i,j}, \forall j \in \{1, 2, 3\}$ 
30   else
   /* Activate full Pooling */
31      $R_j = R_j + R_j^s, \forall j \in \{1, 2, 3\}$ 
32     for  $i \in \mathcal{L}^w$  do
33       if  $r_{i,j} \leq R_j, \forall j \in \{1, 2, 3\}$  then
34          $\mathcal{L}^a \cup i$  and  $x_i = 1$ 
35          $R_j = R_j - r_{i,j}, \forall j \in \{1, 2, 3\}$ 
36     else
37       the slice request  $i$  is rejected

```

V. PERFORMANCE EVALUATION

In this section, we assess the performance of the proposed admission control algorithm, and compare it with a state of the art (SOTA) algorithm [10] that prioritizes requests based on their class priority $C_i, \forall i \in \{1, \dots, 6\}$, in decreasing order and deploys the full pooling mechanism by buying back all the idle resources from already admitted slices. We present the simulation settings in Section V-A, the proposed priority schemes in Section V-B, and some results in Section V-C.

A. Simulation Settings

The simulation scenario consists of an InP having fixed units of three resources : computing $R_1 = 10000$, storage $R_2 = 30000$ and network $R_3 = 50000$. Moreover, the number of requests is generated randomly between 200 and 1300. The nominal priority of each class $C_i, i \in \{1, \dots, 6\}$ correspond to 6, 5, 4, 3, 2, 1, respectively; while the willingness to sell threshold Γ_i is set to 0.6, $\forall i \in \mathcal{N}$.

We introduce a dynamic buyback price that the InP have to pay to the tenant from which it buys back some resources. It is worthy to note that while existing solutions set the buyback price to the same price as initially paid by the tenant or higher [7]-[11]. In this paper, we propose more realistic and consistent model defined as:

$$p'_{i,j} = \frac{d_i^r}{d_i} \times p_{i,j}, \quad (6)$$

where: d_i^r, d_i and $p_{i,j}$ are the remaining validity duration, duration and initial unit price paid by the tenant for that resource, respectively. Note that $0 < p'_{i,j} < p_{i,j}$ and $d_i^r < d_i$.

The willingness to sell probability of a slice is defined as:

$$P_i = \frac{r_{i,j}^u}{r_{i,j}} \times \frac{d_i^u}{d_i}, \quad (7)$$

where : $r_{i,j}^u$ is the amount of resources already used by the request from the total resources allocated to it $r_{i,j}$; d_i^u and d_i are the used duration and total duration of slice i , respectively. This probability indicates to what extent the tenant has used its resources and how long they have been used. The simulation parameters are captured in Table I.

TABLE I: Simulation Parameters.

Parameter	Value(s)
Nominal priority	{6,5,4,3,2,1}
$d_i \forall i \in \mathcal{N}$	random between 1 and 12 months
Willingness to sell threshold (Γ_i)	0.6
Number of requests N	random between 200 and 1300
InP's capacity	$R_1 = 10^3, R_2 = 3.10^3, R_3 = 5.10^3$
$r_{i,1} \forall i \in \mathcal{N}$	random between 1 and 50 units
$r_{i,2} \forall i \in \mathcal{N}$	random between 1 and 100 units
$r_{i,3} \forall i \in \mathcal{N}$	random between 1 and 500 units
$p_{i,j}, j \in \{1, 2, 3\}, i \in \text{class } C_1$	$p_{1,1} = 10, p_{1,2} = 9, p_{1,3} = 8$
$p_{i,j}, j \in \{1, 2, 3\}, i \in \text{class } C_2$	$p_{1,1} = 9, p_{1,2} = 8, p_{1,3} = 7$
$p_{i,j}, j \in \{1, 2, 3\}, i \in \text{class } C_3$	$p_{1,1} = 8, p_{1,2} = 7, p_{1,3} = 7$
$p_{i,j}, j \in \{1, 2, 3\}, i \in \text{class } C_4$	$p_{1,1} = 7, p_{1,2} = 6, p_{1,3} = 5$
$p_{i,j}, j \in \{1, 2, 3\}, i \in \text{class } C_5$	$p_{1,1} = 6, p_{1,2} = 5, p_{1,3} = 5$
$p_{i,j}, j \in \{1, 2, 3\}, i \in \text{class } C_6$	$p_{1,1} = 5, p_{1,2} = 4, p_{1,3} = 3$

B. Proposed priority schemes

The proposed admission control Algorithm 1 is executed given a priority metric P_i which should be defined for each slice request $i \in \mathcal{N}$. Therefore, we propose four metrics, we describe in the following.

1) *Revenue-cost ratio based scheme (RCRS)*: In this scheme, the algorithm employs a dynamic priority defined as a ratio of the revenue to be accrued from accepting a request to the total resources needed by this request.

$$\alpha_i = \frac{U_i}{\sum_{j=1}^3 r_{i,j}}, \quad (8)$$

where: U_i and $\sum_{j=1}^3 r_{i,j}$ are the revenue to be accrued to the InP from admitting the slice request i and the total required resources (computing: $r_{i,1}$, storage: $r_{i,2}$ and network: $r_{i,3}$), respectively.

2) *Revenue based scheme (RS)*: The priority for the revenue based scheme corresponds to the revenue only. It is defined as follows :

$$\alpha_i = U_i. \quad (9)$$

3) *Priority-duration ratio based scheme (PDRS)*: The priority in this scheme is calculated based on the nominal priority C_i and the requests duration as follows :

$$\alpha_i = \frac{C_i}{\frac{1}{N} \sum_{i=1}^N C_i} \times \frac{d_i}{\frac{1}{N} \sum_{i=1}^N d_i} \quad (10)$$

This priority allows one to create a balance between the long term revenue of the InP and fairness to other low priority requests. Indeed, low priority requests with longer duration will have their priority values increased, which will increase their chances of getting accepted as opposed to nominal priority based algorithms.

4) *Class based scheme (CS)*: In this scheme, we use the nominal priority inherited by requests from their class type $C_i, \forall i \in \{1, \dots, 6\}$ like SOTA schemes [4], [5].

$$\alpha_i = C_i, \quad (11)$$

where : $C_i \in \{1, 2, 3, 4, 5, 6\}$.

C. Comparison to state of the art

In this subsection, we present the results showing the performance of the proposed algorithm, implemented with the different priority schemes, PCRS, RS, PDRS, CS, against the state of the art solution [10]. Each value in the results is an average of 1000 simulation runs.

Fig. 2 shows how the revenue earned by the InP from admitting the various slice requests scales with the number of requests N . The results indicate that at low slice requests rate, all the implemented schemes have similar performance. However, as the number of requests increase, the provider exhausts the resources in its resource pool and have to buyback from willing already admitted tenants. The results indicate that from this point, the proposed algorithm, represented by PCRS, RS, PDRS, CS, outperforms the SOTA algorithm. This

improved performance is due to the dynamic priority introduced in the proposed algorithm and further demonstrates the impact of the priority on the revenue of the InP.

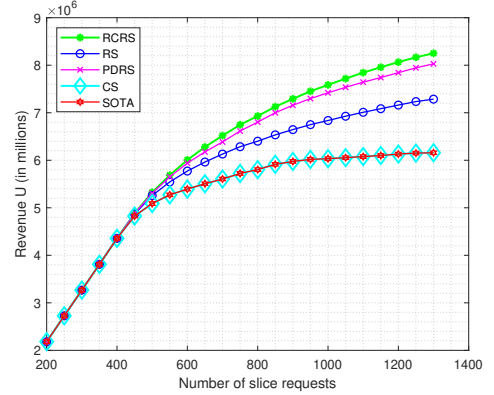


Fig. 2: Comparison in terms of revenue between the proposed algorithm and the SOTA algorithm. We represent for different priority schemes.

Fig. 3 shows the performance in terms of buyback cost. We can see that the proposed algorithm, including PCRS, RS, PDRS, CS, incurs less buyback cost than the SOTA algorithm right from the onset of the buyback process. The reason is that, all the proposed priority schemes employ step-wise pooling which enables the InP to only buyback as much as is required; thus, leading to lower buyback cost while with the SOTA, the InP buys back without recourse to the needed resources.

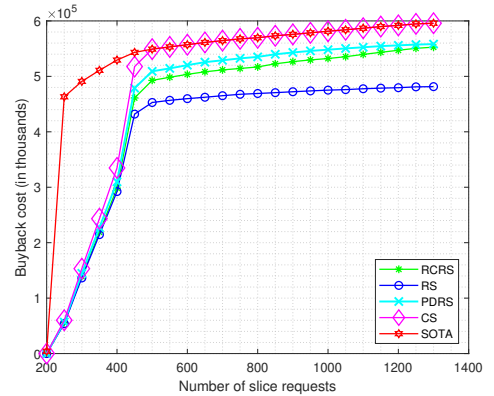


Fig. 3: Comparison in terms of buyback cost between the proposed algorithm and the SOTA algorithm.

Fig. 4 shows the net revenue earned by the InP from admitting the various slice requests. It indicates that at a low slice requests rate, all the schemes have similar performance. However, when the number of requests increases, the provider runs out of resources and initiates a buyback process, resulting in a sharp contrast in the net revenue between the proposed algorithm and SOTA solution. This is due to the dynamic priority and step-wise buyback mechanism introduced in the proposed algorithm, namely PCRS, RS, PDRS, CS, that ensures

that the InP only buys back the units of resources needed as against buying back without recourse to the needed resources. Thus, reducing the provider's buyback cost and consequently increasing the net revenue.

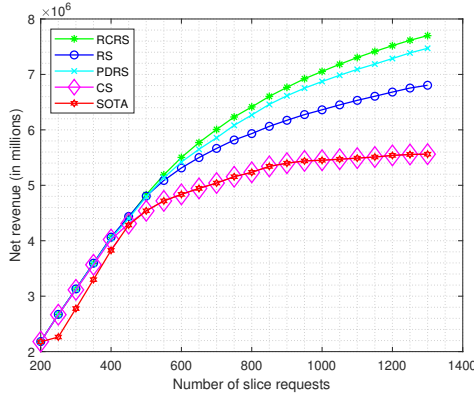


Fig. 4: Comparison in terms of net revenue between the proposed algorithm and the SOTA algorithm.

Fig. 5 depicts the acceptance rate. The results show that the proposed algorithm slightly outperforms the SOTA when it implements the priority schemes PCRS, PDRS, CS. However, implementing RS as a priority scheme provides the lowest acceptance rate. This can be attributed to the role played by the duration and resource units of the requests. That is, requests with higher duration and resource request, will yield higher revenues but also lead to the InP admitting less before running out of resources. Similarly, admitted requests with longer duration will invariably yield higher willingness to sell probability and be thus less willing to sell, making them admit less through the buyback process. This is reflected in the low buyback incurred by the RS.

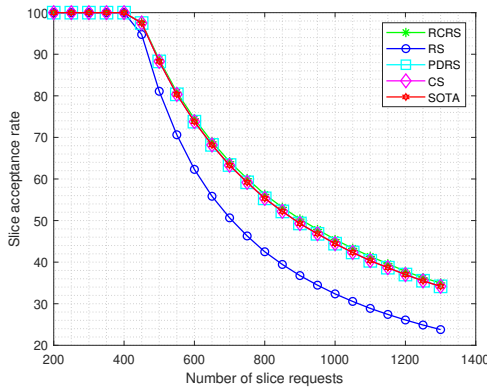


Fig. 5: Comparison in terms of acceptance rate between the proposed algorithm and the SOTA algorithm.

VI. CONCLUSION

In this work, we studied the admission control problem in 5G and beyond networks to determine which slice requests

to accept. We formulated the problem as a multi-objective optimization problem to maximize the revenue and minimize the buyback cost. We proposed an algorithm and different priority schemes, that employ the step-wise pooling mechanism, and dynamic priority to determine the order in which requests should be processed. We evaluated the performance of the proposed algorithm against the SOTA which employs nominal priority and full pooling. The simulation results showed that the proposed algorithm outperforms the SOTA algorithm in terms of InP's revenue and buyback cost.

In future works, we will investigate the impact of the proposed solution on per slice class acceptance and rejection rates with a view to determining its impact on the various class of slices.

ACKNOWLEDGMENT

This work is supported by the Nigerian government through the Petroleum Technology Development Fund (PTDF) overseas merit award scholarship.

REFERENCES

- [1] M. Agiwal, A. Roy and N. Saxena, "Next Generation 5G Wireless Networks: A Comprehensive Survey," in IEEE Communications Surveys 'I&' Tutorials, vol. 18, no. 3, pp. 1617-1655, third quarter 2016, doi: 10.1109/COMST.2016.2532458.
- [2] J. G. Andrews et al., "What Will 5G Be?," in IEEE Journal on Selected Areas in Communications, vol. 32, no. 6, pp. 1065-1082, June 2014, doi: 10.1109/JSAC.2014.2328098.
- [3] M. R. Raza, C. Natalino, P. Ohlen, L. Wosinska and P. Monti, "A Slice Admission Policy Based on Reinforcement Learning for a 5G Flexible RAN," 2018 European Conference on Optical Communication (ECOC), Rome, Italy, 2018, pp. 1-3, doi: 10.1109/ECOC.2018.8535483.
- [4] B. Bordel, R. Alcarria, D. Sánchez-de-Rivera and Á. Sánchez, "An inter-slice management solution for future virtualization-based 5G systems" in Advanced Information Networking and Applications, Cham:Springer, vol. 926, pp. 1059-1070, 2019, doi:10.1007/978-3-030-15032-7_89
- [5] M. O. Ojijo and O. E. Falowo, "A Survey on Slice Admission Control Strategies and Optimization Schemes in 5G Network," in IEEE Access, vol. 8, pp. 14977-14990, 2020, doi: 10.1109/ACCESS.2020.2967626.
- [6] A. El-mekkawi, X. Hesselbach and J. R. Piney, "A Novel Admission Control Scheme for Network Slicing based on Squatting and Kicking Strategies," in Proc. JITEL, 2019, pp. 1-8.
- [7] B. Bakhshi, J. Mangues-Bafalluy and J. Baranda, "R-Learning-Based Admission Control for Service Federation in Multi-domain 5G Networks," 2021 IEEE Global Communications Conference (GLOBECOM), Madrid, Spain, 2021, pp. 1-6, doi: 10.1109/GLOBECOM46510.2021.9685936.
- [8] V. C. Ferreira, H. H. Esmat, B. Lorenzo, S. Kundu and F. Felipe M. G., "Reinforcement Learning based Multi-Attribute Slice Admission Control for Next-Generation Networks in a Dynamic Pricing Environment," 2022 IEEE 95th Vehicular Technology Conference: (VTC2022-Spring), Helsinki, Finland, 2022, pp. 1-5, doi: 10.1109/VTC2022-Spring54318.2022.9860729.
- [9] N. Sen and A. F. A., "Intelligent Admission and Placement of O-RAN Slices Using Deep Reinforcement Learning," 2022 IEEE 8th International Conference on Network Softwarization (NetSoft), Milan, Italy, 2022, pp. 307-311, doi: 10.1109/NetSoft54395.2022.9844089.
- [10] S. Saxena and K. M. Sivalingam, "DRL-Based Slice Admission Using Overbooking" in 5G Networks," in IEEE Open Journal of the Communications Society, vol. 4, pp. 29-45, 2023, doi: 10.1109/OJ-COMS.2022.3227591.
- [11] M. Dai, L. Luo, J. Ren, H. Yu and G. Sun, "PSACCF: Prioritized Online Slice Admission Control Considering Fairness in 5G/B5G Networks," in IEEE Transactions on Network Science and Engineering, vol. 9, no. 6, pp. 4101-4114, 1 Nov.-Dec. 2022, doi: 10.1109/TNSE.2022.3195862.